Structural identification of unate-like genetic network models from time-lapse protein concentration measurements

Porreca, Riccardo ; Cinquemani, Eugenio ; Lygeros, John ; Ferrari-Trecate, Giancarlo

Abstract: We consider the problem of learning dynamical models of genetic regulatory networks from time-lapse measurements of gene expression. In our previous work [1], we described a method for the structural and parametric identification of ODE models that makes use of concurrent measurements of concentrations and synthesis rates of the gene products, and requires the knowledge of the noise statistics. In this paper we assume all these pieces of information are not simultaneously available. In particular we propose extensions of [1] that make the method applicable to protein concentration measurements only. We discuss the performance of the method on experimental data from the network IRMA, a benchmark synthetic network engineered in yeast Saccharomices cerevisiae.

DOI: https://doi.org/10.1109/CDC.2010.5717922

Structural identification of unate-like genetic network models from time-lapse protein concentration measurements

Riccardo Porreca, Eugenio Cinquemani, John Lygeros and Giancarlo Ferrari-Trecate

***Abstract*— We consider the problem of learning dynamical models of genetic regulatory networks from time-lapse measurements of gene expression. In our previous work [1], we described a method for the structural and parametric identification of ODE models that makes use of concurrent measurements of concentrations and synthesis rates of the gene products, and requires the knowledge of the noise statistics. In this paper we assume all these pieces of information are not simultaneously available. In particular we propose extensions of [1] that make the method applicable to protein concentration measurements only. We discuss the performance of the method on experimental data from the network IRMA, a benchmark synthetic network engineered in yeast *Saccharomices cerevisiae*.**

I. Introduction

The regulation of gene expression is central to the ability of living organisms to adapt to their environment, grow, and replicate. It is achieved via a complex network of biochemical interactions among genes, gene products (proteins) and other chemical complexes. Modelling the dynamics of this network allows one to predict the response of the organism to various stimuli (e.g., heat shock and nutrients starvation), with natural impact on applications such as drug development and control of biochemical industrial processes. Various modelling approaches have been developed with success for the identification of the network of interactions [2], [3], [4].

Modern experimental techniques for the time-lapse measurement of gene expression levels allow one to take modelling one step further by the identification of gene expression dynamics. We are interested in the inference of kinetic models [5]. These are ordinary differential equation (ODE) models whose states represent concentrations of gene products and where interactions among genes are quantified by state-dependent synthesis rate functions. In particular, the algebraic structure of the synthesis rates allows one to capture the network of interactions and the logics of gene expression control in a way much similar to Boolean network models [6]. Due to the vastness of this model class, simplifications must be introduced in order to make the identification problem tractable.

A variety of approximation methods have been considered in recent years. Promising results have been obtained by linearization methods [7], [2], [8], [9]. A successful application of universal approximators is provided by [10]. These methods reconstruct the strengths of the interactions among genes, but do not shed light on the logics governing gene activation. Identification approaches based on piecewise affine models allow for the reconstruction of parameters and logics of gene regulatory networks [11], [12]. The approach captures switch-like regulatory interactions that are well approximated by combinations of step functions [13], [14], but is inapplicable to graded gene activation functions (e.g. combinations of smooth sigmoidal functions) where the approximation obtained by step functions is too coarse. This limitation is partly ameliorated by stochastic piecewise affine modelling in [15], but the resulting identification scheme is limited to parameter estimation.

An approach that preserves the form of Boolean-like kinetic models, accounts for sigmoidal activation functions and avoids parsing all possible model structures was proposed in [1] (see also [16]). The method relies on the use of unate functions [17], a class of Boolean gene activation rules that appears to be a comprehensive description of the observable interactions among genes [18]. Within a class of ODE models with unate-like structure, the method achieves structural and parameter identification at an affordable computational cost. However, it must be applied on experimental data where both the gene product concentrations and their synthesis rates are measured over time, and assumes knowledge of the statistics of the observation noise. Although this data can be obtained by several *ad hoc* methods [19], [20], experimental techniques such as RT-PCR or gene-reporter systems do not provide all this information at once, hence the need for suitable data processing [21].

In this paper we investigate the applicability of the identification algorithm [1] to the common situation where only protein concentration measurements are available over time. Based on the standard biological assumption that protein degradation rates are known, we introduce a preliminary step where the missing information is reconstructed from the available data. We turn protein synthesis rate estimation into a deconvolution problem. We comment on the applicability of common deconvolution approaches [22], [23], [24], [25] and then propose a simple algorithm based on smoothing splines and a bootstrapping technique [26] for reconstructing synthesis rate measurements and noise statistics. Finally, the performance of the complete identification algorithm is compared to state-of-the-art reverse engineering algorithms on the network IRMA [27], a synthetic network engineered in yeast *Saccharomices cerevisiae* and proposed as a benchmark

This work was supported in part by the SystemsX.ch research consortium under the project YeastX.

Riccardo Porreca and John Lygeros are with the Institut für Automatik, ETH Zürich, Switzerland.

Eugenio Cinquemani is with the INRIA Grenoble-Rhône-Alpes, Montbonnot, France.

Giancarlo Ferrari-Trecate is with the Dipartimento di Informatica e Sistemistica, Università di Pavia, Italy.

Corresponding author: Riccardo Porreca, email: rporreca@control.ee.ethz.ch

for gene network inference algorithms.

In Section II we review kinetic models with unate structure and in Section III we report concisely the identification algorithm presented in [1]. In Section IV we address the problem of the reconstruction of the missing data and outline the extended version of the identification algorithms. Gene network identification results on the experimental data from the network IRMA are reported in Section V.

II. Kinetic models with unate structure

This and the next section follow closely [1] (see also the supplementary material of [1] for mathematical proofs).

A. Boolean network modelling with unate functions

A natural qualitative approach to gene regulatory network modelling is provided by Boolean networks [6]. For a network of n genes, the expression status of each gene is encoded by a Boolean variable X_i, with $i = 1, \dots, n$. When $X_i = 0$, the gene is not expressed. When $X_i = 1$, the gene is expressed, i.e. the corresponding product is synthesized. The regulation (activation/inactivation) of gene expression is modeled by a Boolean function $X_i^+ = B_i(X)$, with $X = (X_1, \dots, X_n)$, where X_i^+ denotes the status of gene i at a subsequent time instant. Thus $B_i(X)$ is a rule that captures the network structure and the logics behind (discrete-time) gene expression dynamics. Among all possible Boolean rules, unate (a.k.a. sign-definite) functions [17] were argued to capture the large majority of gene activation rules [18]. They have the property of being monotone with respect to each input variable. When expressed in minimal conjunctive normal form, unate functions have the property that each input variable X_j appears in $B(X)$ either as is (the regulator acts as an activator) or in the negated form $\neg X_j$ (the regulator acts as a repressor), but not both.

B. Kinetic network models with unate structure

In the context of ODE models, it is possible to include the logics of unate activation rules by an appropriate algebraic reformulation of $B_i(X)$ [28]. Let $x_i \in \mathbb{R}_{\geq 0}$, $i = 1, \dots, n$, denote the concentration of the product of gene i, and let $x = \big(x_1, \dots, x_n\big)$. Each x_i is assumed to follow the model

$$\dot{x}_i = g_i(x) - \gamma_i(x), \tag{1}$$

$$g_i(x) = \kappa_{0,i} + \kappa_{1,i} b_i(x), \tag{2}$$

where $g_i(x) \geq 0$ and $\gamma_i(x) \geq 0$ are the synthesis and the degradation rates of the product of gene i, $\kappa_{0,i} \in \mathbb{R}_{\geq 0}$ and $\kappa_{1,i} \in \mathbb{R}_{\geq 0}$ are constants and $b_i(x) : \mathbb{R}^n_{\geq 0} \to [0, 1]$ encodes the logic of the regulation of gene i. At this stage, we are not concerned with the form of $\gamma_i(x)$. In general [5], $b_i(x)$ is a combination (weighted sums and products) of so-called Hill functions [29], [30], [31], i.e. sigmoid-shaped functions

$$\sigma^+(x_j) = \frac{x_j^d}{x_j^d + \eta^d}, \qquad \sigma^-(x_j) = 1 - \sigma^+(x_j) = \frac{\eta^d}{x_j^d + \eta^d},$$

modelling switch-like biochemical interactions, where $d \geq 1$ is a cooperativity coefficient and $\eta > 0$ is a threshold parameter. For the case of unate functions, following [28], $b_i(x)$ is obtained from $B_i(X)$ by replacing X_i by $\sigma^+(x_i)$, logical negation $\neg(\cdot)$ by algebraic complementation $1 - (\cdot)$ and logical conjunctions by products. This yields

$$b_i(x) = \prod_{l=1}^{n_i} \tau_l(x), \qquad \tau_l(x) = 1 - \prod_{j \in J_l} \big(1 - \sigma^{\pm}(x_j)\big), \tag{3}$$

where each J_l is a nonempty subset of $\{1, \dots, n\}$, and each term $\sigma^{\pm}(x_j)$ is uniquely defined as $\sigma^+(x_j)$ or as $\sigma^-(x_j)$. We will refer to (1)–(3) as a kinetic model with unate structure.

C. Sign patterns and related properties

Let us focus on the model (1)–(3) for a single gene i and drop the subscript i from the notation. We define the *sign pattern* $p = (p_1, \dots, p_n) \in \{-1, 0, +1\}^n$ of $g(x)$ (equivalently, of $b(x)$) as follows: for $j = 1, \dots n$,

$$p_j = \begin{cases} 0, & \text{if } j \notin J_l,\ l = 1, \dots, n_i, \\ 1, & \text{if } \sigma^{\pm}(x_j) = \sigma^+(x_j), \\ -1, & \text{if } \sigma^{\pm}(x_j) = \sigma^-(x_j). \end{cases}$$

The *complexity* $C(p)$ of a sign pattern p is defined as the number of nonzero entries of p. Together with the index sets J_l, p defines the *structure* of the model, i.e. the specific form of (3). The family of model structures corresponding to a sign pattern p is denoted by $S(p)$. Let $g(x|p)$ denote a synthesis rate g with sign pattern p. It is easily seen that p encodes the monotonicity properties of $g(x)$. If $p_j = 1$ (resp. $p_j = -1$) then $g(x)$ is monotonically increasing (resp. decreasing) in x_j, while $g(x)$ is independent of x_j if $p_j = 0$. In general, given two vectors $x^1, x^2 \in \mathbb{R}^n_{\geq 0}$, it holds

$$\big[p_j(x_j^2 - x_j^1) \geq 0,\ \forall j\big] \Rightarrow \big[g(x^2|p) - g(x^1|p) \geq 0\big]. \tag{4}$$

Consider a set of m concentration and synthesis rate data pairs (x^k, g^k), with $g^k = g(x^k)$. In the light of (4), a sign pattern is declared *inconsistent* (with the data) if there exist $k, l \in \{1, \dots, m\}$ such that

$$\big[p_j(x_j^k - x_j^l) \geq 0,\ j = 1, \dots, n\big] \text{ and } \big[g^k - g^l < 0\big]. \tag{5}$$

If p is not inconsistent then it is called consistent. It is possible to define a partial order relation $\sqsubseteq$ on sign patterns as follows: $p' \sqsubseteq p$ if and only if $p'_i = p_i,\ \forall i : p'_i \neq 0$. p' is called a subpattern of p and p a superpattern of p'. Every superpattern of a consistent pattern is also consistent and every subpattern of an inconsistent pattern is also inconsistent. Based on (5) and suitable processing of all data pairs (x^k, g^k) and (x^l, g^l), it is possible to define a set $\bar{P}$ such that all inconsistent sign patterns are subpatterns of at least one element of $\bar{P}$. From this, it is possible to derive a set P^* of minimal (with respect to $\sqsubseteq$) consistent sign patterns. All consistent sign patterns are superpatterns of at least one element of P^*, i.e. they form a hierarchy $\mathscr{H}(P^*)$.

III. Identification with complete data

A. Problem statement

For $k = 1, \dots, m$ and $i = 1, \dots, n$, we are given noisy observations of gene product concentrations, $\tilde{x}_i^k$, and

synthesis rates, $\tilde{g}_i^k$, obeying the measurement model

$$\begin{aligned} \tilde{x}_i^k &= x_i^k + e_i^k, \quad x_i^k = x_i(t_k), \\ \tilde{g}_i^k &= g_i^k + \epsilon_i^k, \quad g_i^k = g_i\big(x(t_k)\big), \end{aligned} \tag{6}$$

where $t_1, \ldots, t_k$ is a sequence of measurement instants, e_i^k and ϵ_i^k are mutually uncorrelated Gaussian random variables with zero mean. Their variance, $v_e(x_i^k) = \text{var}(e_i^k)$ and $v_\epsilon(g_i^k) = \text{var}(\epsilon_i^k)$, is a known (smooth) function of x_i^k and of g_i^k. As an example, if v_e and v_ϵ are constant, then additive noise models are obtained. Moreover, $v_e(x_i)$ and $v_\epsilon(x_i)$ that are linear functions of x_i^2 and g_i^2 result in multiplicative noise models. Our objective is to identify structure and parameters of the "simplest" model $g(x)$ of the form (2)–(3) that explains the data in a "statistically acceptable" way. By simplest we mean the model whose sign pattern has minimal complexity. The meaning of "statistically acceptable" will be clarified below. We accept the possibility that a pool $\mathscr{P}$ of models of equal complexity, rather than a single model, is found. The problem being identical for all genes i, we drop this index from the notation wherever no ambiguities arise.

B. Identification algorithm

Since both x and $g(x)$ are observed, the identification problem becomes a nonlinear regression problem. In particular, the degradation term $\gamma(x)$ in (1) does not play any role here. A natural solution is to search the family of candidate models by increasing levels of complexity until a model is found that fits the data for an appropriate choice of all model parameters θ (i.e. κ_0, κ_1 and the parameters η and d for all the sigmoids in the model). Since this family is vast, the idea is to a priori exclude from the search all models $g(x|p)$ whose sign pattern p is inconsistent with the data. This ensures major computational savings and leads to Algorithm 1.

On the basis of the properties in Section II-C, Step 1 computes the set of minimal consistent sign patterns P^*. Since the data is noisy, the condition (5) used to define inconsistent sign patterns is checked in Step 1.I via a standard statistical test on the mean of two Gaussian random variables. Parameter N determines the confidence level of this test (the standard 95% confidence level is obtained for $N = 2$). We defer the reader to [1] for further details.

In Step 2 we seek models with structure compatible with $\mathscr{H}(P^*)$ that explain the data with sufficient accuracy. The search is conducted by increasing levels of complexity ℓ, starting from the simplest models in P^*, and is stopped at the level of complexity where at least one model structure is found for which, after minimization with respect to θ, the regression error δ is smaller than $\tau(\alpha)$. As shown in [1], for an appropriate choice of the regression weights w_k depending on the noise variance functions v_e and v_ϵ, the residual error δ associated to the model with true structure and parameters approximately follows the probability distribution $F_{m-|\theta|}$ of a χ^2 random variable with $m-|\theta|$ degrees of freedom. Then, for $\tau(\alpha) = F^{-1}_{m-|\theta|}(\alpha)$, the condition $\delta < \tau(\alpha)$ corresponds to a statistical test that accepts the true model (rejects false models) with tunable confidence level α.

Algorithm 1 Two-step identification

Step 1. (Selection of consistent model structures)

I. Set $\bar{P} = \emptyset$. For all indices $k, l \in \{1, \ldots, m\}$, if $\tilde{g}_i^k - \tilde{g}_i^l < -N\sqrt{v_\epsilon(\tilde{g}_i^k) + v_\epsilon(\tilde{g}_i^l)}$ then compute

$$\bar{p}_j = \begin{cases} -1, & \text{if } \tilde{x}_j^k - \tilde{x}_j^l \le -N\sqrt{v_e(\tilde{x}_j^k) + v_e(\tilde{x}_j^l)}, \\ 1, & \text{if } \tilde{x}_j^k - \tilde{x}_j^l \ge N\sqrt{v_e(\tilde{x}_j^k) + v_e(\tilde{x}_j^l)}, \\ 0, & \text{otherwise}, \end{cases}$$

for $j = 1, \ldots, n$, and include $\bar{p} = (\bar{p}_1, \ldots, \bar{p}_n)$ in $\bar{P}$.

II. Compute P^* as the set of minimal patterns p such that no element of $\bar{P}$ is a superpattern of p.

Step 2. (Identification of best consistent models)

Set $\mathscr{P} = \emptyset$. For $\ell = \min\{C(p^*) : \ p^* \in P^*\}$ to n:

V. Generate patterns p such that $C(p) = \ell$ and $p^* \sqsubseteq p$ for some $p^* \in P^*$. For each such p, execute VI.

VI. For all $s \in S(p)$, fit the model $g_i(\cdot)$ with structure s by solving the nonlinear regression problem

$$\delta = \min_\theta \sum_{k=1}^{m} w_k \big(\tilde{g}_i^k - g_i(\tilde{x}^k)\big)^2.$$

If $\delta < \tau(\alpha)$, include the fitted model in $\mathscr{P}$.

VII. If $\mathscr{P} \neq \emptyset$ return $\mathscr{P}$ and exit.

Thanks to the hierarchical search, the procedure favors simple models over complex models. In practice, similar to other gene network reconstruction algorithms [27], an upperbound to the model complexity can be used to ensure that models of unreasonable complexity are not explored. Since the hypothesis $b(x) \equiv 1$ is not explicitly accounted for, prior to the execution of the algorithm, a simple statistical test is recommended in [1] to verify that a constant synthesis rate does not explain the data.

C. Exploiting a priori knowledge

In practice, searching all unate structures $S(p)$ associated with a sign pattern p may be prohibitive. It is usually the case that additional information on the likely structure of the model is available. For illustration purposes, we follow the work of [32] and references therein, showing that a large part of the known gene activation rules fall in a subclass of hierarchically (or nested) canalizing functions [33]. Note that hierarchically canalizing functions are a proper subset of unate functions. In this case, the structure of the model simplifies to

$$b_i(x) = \begin{cases} \sigma^{\pm}(x_{j_1})\sigma^{\pm}(x_{j_2})\sigma^{\pm}(x_{j_3}) \cdots \sigma^{\pm}(x_{j_\ell}) & \text{or} \\ \big[1 - \big(1 - \sigma^{\pm}(x_{j_1})\big)\big(1 - \sigma^{\pm}(x_{j_2})\big)\big] \times & \\ \qquad \sigma^{\pm}(x_{j_3}) \cdots \sigma^{\pm}(x_{j_\ell}), & \end{cases} \tag{7}$$

where ℓ is the number of effective inputs of $b_i(x)$ and $j_1, \ldots, j_\ell$ are pairwise different indices from the set $\{1, \ldots, n\}$. Clearly the model is in the form (3) and $b_i(x)$ corresponds to Boolean functions composed by all AND operators or all AND but one OR between two variables. Moreover ℓ is the complexity of the associated sign pattern.

IV. Estimation of the synthesis rate from concentration measurements

In order to cope with the lack of synthesis rate measurements and variances $v_e(x_i^k)$, we rely on model (1) and consider the case where protein degradation is spontaneous (unregulated). Therefore (1) becomes

$$\dot{x}_i = g_i(x) - a_i x_i, \quad i = 1, \dots, n, \tag{8}$$

where degradation parameters $a_i > 0$ are known. In practice, these parameters are routinely estimated by dedicated biological experiments based on western or northern blotting. Since $g_i(x(t))$ acts as an input to the first-order system (8), the values g_i^k, $k = 1, \dots, m$, can be estimated by solving a *deconvolution* problem with discrete data $\tilde{x}_i^k$. Deconvolution problems are a class of inverse problems that have been studied since the 70's. Due to their inherent ill-posedness, state-of-the-art approaches are based on regularization techniques [22], [23], [34] or, equivalently, on Bayesian estimation [24], [25]. In these methods, the unknown input is found by minimizing a cost functional given by the sum of two terms: the least-square fit and a smoothing term that penalizes irregular estimates. The trade-off between data fit and smoothing is controlled by a positive regularization parameter. In the Bayesian setting, the optimal value of the regularization parameter depends on the covariance of the Gaussian prior modeling the input and the variances $v_\epsilon(g_i^k)$. When these quantities are not known one has to tune the regularization parameter using data-based techniques such as Generalized Cross Validation (GCV) [35] or Maximum-Likelihood (ML) [36], [24]. Quite remarkably, the ML methods proposed in [24] are also capable of providing a posteriori estimates of the variances $v_\epsilon(g_i^k)$ and $v_e(x_i^k)$, hence providing all pieces of information required by Algorithm 1 in Section III-B.

Unfortunately, the application of the deconvolution algorithms in [24] to the IRMA data (see Section V) resulted in severe over- or under-smoothing for different gene products. Moreover, similar remarks apply to the use of GCV for tuning the regularization parameter. This suggests that noise and input models assumed by GCV and ML are not well suited for our test case. The best results have been found using the simpler deconvolution algorithm described in the next section together with a bootstrap technique for obtaining estimates of $v_\epsilon(g_i^k)$ and $v_e(x_i^k)$.

A. Estimation based on smoothing splines and bootstrap resampling

Given noisy measurements $\tilde{x}_i^k$ of $x_i(t_k)$, a simple approach to deconvolution consists in estimating the function $x_i(t)$, $t \geq 0$ by means of a smoothing spline $\hat{x}_i(t)$, whose derivative $\dot{\hat{x}}(t)$ can be computed analytically [35]. Then, $g_i(t)$ is estimated as

$$\hat{g}_i(t) = \dot{\hat{x}}_i(t) + a_i \hat{x}_i(t) \ . \tag{9}$$

In order to obtain some meaningful statistics about the estimate of g_i, the bootstrap method [26] of residual resampling is applied. Such method can be used in general to infer the distribution of any mathematical transformation of a smoothing spline (or any other regression curve). The idea is to resample with replacement the residuals of the spline (under the assumption that they are i.i.d.) in order to generate several new noisy datasets. For each of such datasets, a new smoothing spline is computed along with the mathematical transformation of interest. This allows one to obtain a distribution of the desired quantities. For instance, the residual resampling method has been applied in the analysis of reporter gene measurements [21]. In our context, mean and variance of the bootstrap replicates of $\hat{g}_i(t_k)$ are taken as estimates of g_i^k and $v_\epsilon(g_i^k)$ to be used in the identification procedure. The method is reported in Algorithm 2, where a generalization of residuals resampling is considered in order to allow for different weights w^k associated to the residual $\tilde{x}_i^k - \hat{x}_i(t_k)$ in the spline computation. Such weights are typically associated with the relative uncertainty of the data, thus making it possible to apply the method under the desired noise model (e.g., additive or multiplicative noise). We highlight that no direct information about the absolute uncertainty (noise variance) of the datapoints is needed. Rather, it is the distribution of the resampled residuals that provides such information, thus allowing the estimation of $v_e(x_i^k)$ as well. The proposed method only requires to specify the type of spline to be used (e.g., a cubic spline), the number N_r of resampling iterations and the smoothing parameter that makes the spline ranging from an interpolant curve to a simple polynomial.

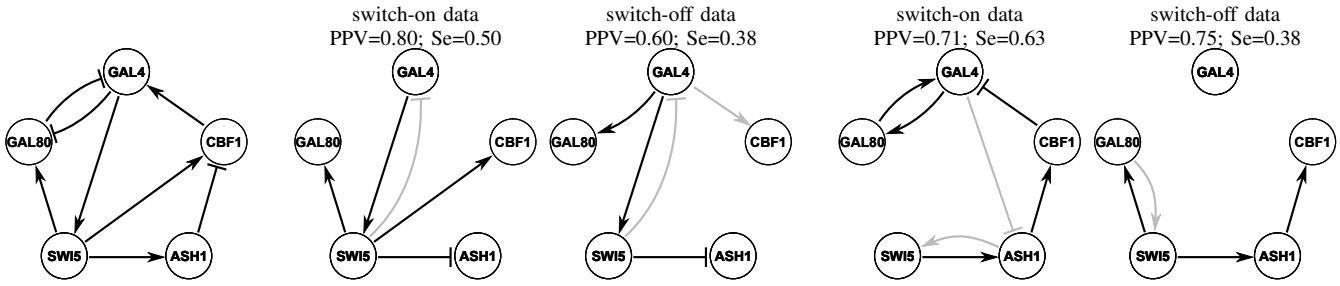
Algorithm 2 Bootstrap spline-based resampling.

1: compute the spline $\hat{x}_i(t)$ from $\{\tilde{x}_i^k\}$ using weights $\{w^k\}$
2: let $R = \{w^k(\tilde{x}_i^k - \hat{x}_i(t_k)), k = 1, \dots, m\}$
3: **for** $r = 1$ to N_r **do**
4: extract with replacement m residuals $\{\varepsilon^k\}$ from R
5: let $\tilde{x}_i^{k(r)} = \hat{x}_i(t_k) + \varepsilon^k / w^k, k = 1, \dots, m$
6: compute the spline $\hat{x}_i^{(r)}(t)$ from $\{\tilde{x}_i^{k(r)}\}$ using weights $\{w^k\}$
7: let $\hat{g}_i^{k(r)} = \dot{\hat{x}}_i^{(r)}(t_k) + \gamma_i \hat{x}_i^{(r)}(t_k), k = 1, \dots, m$
8: **end for**
9: let $\hat{g}_i^k = \frac{1}{N_r} \sum_r \hat{g}_i^{k(r)}$, $\hat{v}_\epsilon(g_i^k) = \frac{1}{N_r - 1} \sum_r (\hat{g}_i^k - \hat{g}_i^{k(r)})^2$ and $\hat{v}_\epsilon(x_i^k) = \frac{1}{m-1} \sum_{\varepsilon \in R} (\varepsilon / w^k)^2$

V. Results on the IRMA network

We now discuss the application of the identification algorithm combined with the estimation of the synthesis rates on IRMA, a synthetic network engineered in *Saccharomyces cerevisiae* cells and proposed as a benchmark for reverse engineering algorithms [27]. The IRMA network comprises five genes and its graphical representation is reported in Fig. 1a. Time-series of gene product concentrations were collected *in vivo* under two growth-medium conditions called *switch-on* and *switch-off*. In particular, 15 and 20 datapoints collected every 20 and 10 minutes are available for the switch-on and switch-off experiments, respectively. A more detailed description of the used dataset can be found in the supplementary material of [1].

A comparison of the performance of various state-of-the-art network reconstruction techniques, ranging from ODE



(a) True network (b) TSNI (c) Algorithm 1–2

Fig. 1: (a) True network of interactions in IRMA. Results obtained by (b) the TSNI algorithm [27] and by (c) Algorithms 1 and 2. Gray edges denote incorrect direction of the inferred interactions.

models to Bayesian and information theoretic approaches, is provided in [27]. Performance is assessed by comparing the *unsigned* directed graph produced by each method to the unsigned version of the graph in Fig. 1a. In particular, Positive Predictive Value PPV=TP/(TP+FP) (TP=True Positive edges, FP=False Positive edges) and Sensitivity Se=TP/(TP+FN) (FN=False Negative edges) were used as performance measures. Notice that unsigned graphs make no distinction between activatory and inhibitory interactions. According to the study in [27], the TSNI algorithm [37], based on linearized ODE models, was able to achieve the best performance in the context of reverse-engineering from time-series data. Its results and performance are illustrated in Fig. 1b. and will be used for comparison in our study. Here we focus on the identification of the network structure and briefly comment on the value of the estimated model for simulation. Parameters in TSNI and in our method have very different meanings and therefore they will not be compared.

Preliminary tests on the application of our method showed that, due to the scarcity of the data set and the high level of noise, for every gene i, the decay factor $\gamma_i x_i$ (used for reconstructing g_i) cannot be distinguished from an autoregulation function $\sigma^{\pm}(x_i)$. The same problem is inherent in TSNI where, due to linearization, autoregulation and degradation term both contribute to the i-th diagonal element of the network interaction matrix and are hence non-distinguishable (see [37] for further details). Therefore, we made the assumption that $\sigma^{\pm}(x_i)$ does not appear in g_i, thus excluding autoregulation.

In order to apply Algorithm 1 to the IRMA datasets, we first estimated the synthesis rate values by means of Algorithm 2, where $N_r = 1000$ resampling steps were performed. For the degradation rates a_i we used the values estimated in [27] from a different dataset. We considered the standard cubic smoothing splines provided by the MATLAB function `csaps`. The smoothing parameter was tuned empirically, following the guidelines in the Matlab Spline Toolbox manual [38], in order to provide a good compromise between data fit and smoothing effect on all time series. No specific assumptions were made about the data uncertainty, i.e. we employed unitary weights w^k for all measurements. Then, Algorithm 1 was applied with $N = 3$ and $\alpha = 0.95$.

Results and performance of the application of Algorithms 2 and 1 are summarized in Fig. 1c. The performance measures are comparable to those obtained by TSNI. In particular, PPV values tend to favor a different method for each dataset, while both our method and TSNI introduce 3 spurious edges in total. On the other hand, the values of Se slightly favor Algorithm 1–2 over TSNI. This is also supported by the fact that our method is able to recover 6 existing interactions (edges) while TSNI only recovers 5. Finally, note that differently from TSNI, which only aims at reconstructing the topology of the gene network, our method produces models that can be used for simulating the network behavior over time. In particular, the acceptance criterion in step VI of Algorithm 1 guarantees that reconstructed models are statistically significant. This is confirmed by the relatively good fit (compared to the noise affecting both concentrations an synthesis rates) of the estimated model to the data, as shown in Fig. 2 for the switch-on time series. Note that, despite the relevant estimates of the interactions among genes, the estimates of the interaction signs are inaccurate. Since the estimated model is statistically relevant and provides a good fit (see Fig. 2), we argue that the network structure of Fig. 1a and the estimated structure of Fig. 1c cannot be discriminated from the given experimental dataset.

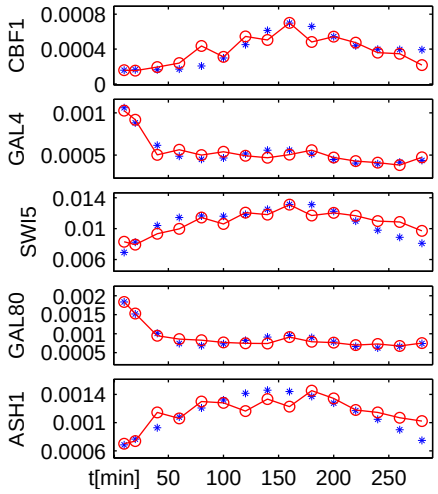


Fig. 2: Synthesis rates $\hat{g}_i^k$, $i = 1, \ldots, 5$, estimated by Algorithm 2 for the IRMA switch-on dataset (stars). Circles denote the values predicted by the functions $g_i(x)$ reconstructed by Algorithm 1.

VI. Conclusions

In this paper we generalized the gene-network reverse-engineering algorithm proposed in [1] to the case where only gene product concentration measurements are available. For the case of unregulated protein degradation, we showed that the synthesis rates, their variance and the variance of gene product concentrations can be estimated through a deconvolution algorithm exploiting bootstrapping. We commented on the limits of the available deconvolution methods and proposed an alternative simple approach based on smoothing splines. The algorithm has been tested on experimental data available for the IRMA network and is capable of achieving a level of performance comparable to the TSNI method.

We observed a limited overlap in the set of correct interactions discovered by TSNI and by our method, suggesting that both methods can provide valuable hints on the network topology. Differently from TSNI, our procedure infers dynamical models that can also be used for simulating the system over time, and whose validity is quantified by the confidence level of a statistical acceptance test.

Future investigations include the evaluation of our method on data from gene reporter systems, an experimental technique ensuring high-quality high-density time-lapse datasets and accompanied by effective data preprocessing tools [39].

References

[1] R. Porreca, E. Cinquemani, J. Lygeros, and G. Ferrari-Trecate, "Identification of genetic network dynamics with unate structure," *Bioinformatics*, vol. 26, pp. 1239–1245, 2010.

[2] M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. di Bernardo, "How to infer gene networks from expression profiles," *Mol. Syst. Biol.*, vol. 3, p. 78, 2007.

[3] T. Gardner and J. Faith, "Reverse-engineering transcription control networks," *Phys. Life Rev.*, vol. 2, no. 1, pp. 65–88, 2005.

[4] F. Markowetz and R. Spang, "Inferring cellular networks: A review," *BMC Bioinform.*, vol. 28, no. Suppl. 6, p. S5, 2007.

[5] H. de Jong, "Modeling and simulation of genetic regulatory systems: A literature review," *J. Comput. Biol.*, vol. 9, no. 1, pp. 69–105, 2002.

[6] S. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *J. Theor. Biol.*, vol. 22, pp. 437–467, 1969.

[7] M. Zavlanos, A. Julius, S. Boyd, and G. Pappas, "Identification of stable genetic networks using convex programming," in *Proc. American Control Conference*, 2008, pp. 2755–2760.

[8] T. Gardner, D. di Bernardo, D. Lorenz, and J. Collins, "Inferring genetic networks and identifying compound mode of action via expression profiling," *Science*, vol. 301, no. 5629, pp. 102–105, 2003.

[9] E. Cinquemani, A. Milias-Argeitis, S. Summers, and J. Lygeros, *Local identification of piecewise deterministic models of genetic networks*. Springer, 2009, vol. N.5469 of the LNCS Series, pp. 105–119.

[10] J. Jaeger, S. Surkova, M. Blagov, H. Janssens, D. Kosman, K. Kozlov, Manu, E. Myasnikova, C. Vanario-Alonso, M. Samsonova, D. Sharp, and J. Reinitz, "Dynamic control of positional information in the early *Drosophila* embryo," *Nature*, vol. 430, no. 6997, pp. 368–371, 2004.

[11] S. Drulhe, G. Ferrari-Trecate, and H. de Jong, "Reconstruction of switching thresholds in piecewise-affine models of genetic regulatory networks," *IEEE Trans. Automat. Control*, vol. 53, no. 1, pp. 153–165, 2008.

[12] R. Porreca, S. Drulhe, H. de Jong, and G. Ferrari-Trecate, "Structural identification of piecewise-linear models of genetic regulatory networks," *J. Comput. Biol.*, vol. 15, pp. 1365–1380, 2008.

[13] L. Glass and S. Kauffman, "The logical analysis of continuous, nonlinear biochemical control networks," *J. Theor. Biol.*, vol. 39, no. 1, pp. 103–129, 1973.

[14] H. de Jong, J.-L. Gouze, C. Hernandez, M. Page, T. Sari, and J. Geiselmann, *Hybrid modeling and simulation of genetic regulatory networks: A qualitative approach*. Berlin: Springer–Verlag, 2003, vol. N.2623 of the LNCS Series, pp. 267–282.

[15] E. Cinquemani, A. Milias-Argeitis, S. Summers, and J. Lygeros, "Stochastic dynamics of genetic networks: modelling and parameter identification," *Bioinformatics*, vol. 24, no. 23, pp. 2748–2754, 2008.

[16] E. Cinquemani, R. Porreca, J. Lygeros, and G. Ferrari-Trecate, "Canalizing structure of genetic network dynamics: modelling and identification via mixed-integer programming," *Proc. 48th IEEE Conference on Decision and Control*, pp. 5618–5623, 2009.

[17] J. Aracena, "Maximum number of fixed points in regulatory boolean networks," *Bull. Math. Biol.*, vol. 70, pp. 1398–1409, 2008.

[18] J. Grefenstette, S. Kim, and S. Kauffman, "An analysis of the class of gene regulatory functions implied by a biochemical model," *BioSystems*, vol. 84, pp. 81–90, 2006.

[19] M. Ronen, R. Rosenberg, B. I. Shraiman, and U. Alon, "Assigning numbers to the arrows: Parameterizing a gene regulation network by using accurate expression kinetics," *PNAS*, vol. 99, no. 16, pp. 10 555–10 560, Aug. 2002.

[20] D. Brown and C. P. Lostroh, "Inferring gene expression dynamics from reporter protein levels," *Biotechnol. J.*, vol. 3, pp. 1437–1448, 2008.

[21] H. de Jong, C. Ranquet, D. Ropers, C. Pinel, and J. Geiselmann, "Experimental and computational validation of models of fluorescent and luminescent reporter genes in bacteria," *BMC Systems Biology*, vol. 4, no. 1, p. 55, 2010.

[22] A. Tikhonov and V. Arsenin, *Solutions of ill-posed problems*. Washington: Winston/Wiley, 1977.

[23] M. Bertero, C. De Mol, and E. Pike, "Linear inverse problem with discrete data I: General formulation and singular system analysis," *Inverse Problems*, vol. 1, pp. 301–330, 1985.

[24] G. De Nicolao, G. Sparacino, and C. Cobelli, "Non parametric input estimation in physiological system: problems, methods and case studies," *Automatica*, vol. 33, pp. 851–870, 1997.

[25] G. Ferrari-Trecate and G. De Nicolao, "Regularization networks for inverse problems: A state-space approach," *Automatica*, vol. 39, pp. 669–676, 2003.

[26] L. C. Hamilton, *Regression with graphics : a second course in applied statistics*. Duxbury Press, 1991.

[27] I. Cantone, L. Marucci, F. Iorio, M. A. Ricci, V. Belcastro, M. Bansal, S. Santini, M. di Bernardo, D. di Bernardo, and M. P. Cosma, "A yeast synthetic network for in vivo assessment of reverse-engineering and modeling approaches," *Cell*, vol. 137, no. 1, pp. 172–181, 2009.

[28] E. Plahte, T. Mestl, and S. Omholt, "A methodological basis for description and analysis of systems with complex switch-like interactions," *J. Math. Biol.*, vol. 36, pp. 321–348, 1998.

[29] H. Yang, C. Hsu, and M. Hwang, "An analytical rate expression for the kinetics of gene transcription mediated by dimeric transcription factors," *J. Biochem.*, vol. 142, pp. 135–144, 2007.

[30] A. D. Keller, "Model genetic circuits encoding autoregulatory transcription factors," *J. Theor. Biol.*, vol. 172, no. 2, pp. 169–185, 1995.

[31] A. Becskei, B. Seraphin, and L. Serrano, "Positive feedback in eukaryotic gene networks: cell differentiation by graded to binary response conversion," *EMBO J.*, vol. 20, pp. 2528–2535, 2001.

[32] S. Nikolajewa, M. Friedel, and T. Wilhelm, "Boolean networks with biologically relevant rules show ordered behavior," *BioSystems*, vol. 90, pp. 40–47, 2007.

[33] A. S. Jarrah, B. Raposa, and R. Laubenbacher, "Nested canalyzing, unate cascade, and polynomial functions," *Physica D.*, vol. 233, no. 2, pp. 167–174, 2007.

[34] A. Aswani, P. Bickel, and C. Tomlin, "Regression on manifolds: Estimation of the exterior derivative," *Ann. Stat.*, 2010, to appear.

[35] G. Wahba, *Spline Models for Observational Data*. SIAM, Philadelphia, 1990.

[36] W. Wecker and C. Ansley, "The signal extraction approach to nonlinear regression and spline smoothing," *J. Amer. Statist. Assoc.*, vol. 78, pp. 81–89, 1983.

[37] M. Bansal and D. di Bernardo, "Inference of gene networks from temporal gene expression profiles," *IET Syst. Biol.*, vol. 1, no. 5, pp. 306–312, 2007.

[38] *Spline Toolbox 3 User's Guide*, The MathWorks Inc., 2009.

[39] F. Boyer, B. Besson, G. Baptist, J. Izard, C. Pinel, D. Ropers, J. Geiselmann, and H. de Jong, "WellReader: A MATLAB Program for the Analysis of Fluorescence and Luminescence Reporter Gene Data," *Bioinformatics*, 2010.